

METHOD FOR MANAGING FILE AND RECORDING OR REPRODUCING DEVICE

Publication number: JP2001325128 (A)

Publication date: 2001-11-22

Inventor(s): MATSUURA YOKO; KUBOTA YOSHIYASU; TAHIRA TAKEHIKO

Applicant(s): SONY CORP

Classification:

- international: **G06F12/00; G06F12/00; (IPC1-7): G06F12/00**

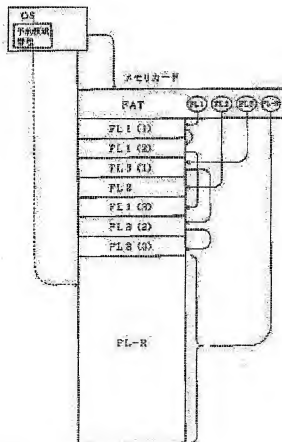
- European:

Application number: JP20000147480 20000515

Priority number(s): JP20000147480 20000515

Abstract of JP 2001325128 (A)

PROBLEM TO BE SOLVED: To realize a file management method suitable for audio data or the like. **SOLUTION:** A physically continued continuous recording space (FL-R) is secured in a recording medium whose data file is managed by a 1st file system such as an FAT so that the whole continuous recording space is managed by the 1st file system and a 2nd file system is arranged in the continuous recording space so that the recording/reproducing of the data file in the continuous recording space is managed by the 2nd file system.



(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-325128

(P2001-325128A)

(43)公開日 平成13年11月22日(2001.11.22)

(51)Int.Cl.

G 0 6 F 12/00

識別記号

5 0 1

F I

C 0 6 F 12/00

データ・ド (参考)

5 0 1 H 5 B 0 8 2

審査請求 未請求 請求項の数 5 O L (全 24 頁)

(21)出願番号 特願2000-147480(P2000-147480)

(22)出願日 平成12年5月15日(2000.5.15)

(71)出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72)発明者 松浦 陽子

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(72)発明者 久保田 芳義

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(74)代理人 100086841

弁理士 脇 篤夫

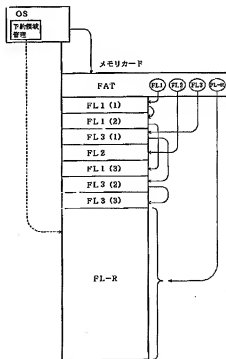
最終頁に続く

(54)【発明の名称】 ファイル管理方法、記録又は再生装置

(57)【要約】

【課題】 オーディオデータ等に好適なファイル管理方法の実現。

【解決手段】 F A T等の第1のファイルシステムでデータファイルが管理される記録媒体内に、物理的に連続した連続記録空間 (F L-R) を確保して、該連続記録空間の全体を上記第1のファイルシステムで管理されるようにすると共に、上記連続記録空間内に第2のファイルシステムを配し、上記連続記録空間内でのデータファイルの記録再生が、上記第2のファイルシステムにより管理されるようにする。



【特許請求の範囲】

【請求項1】 第1のファイルシステムでデータファイルが管理される記録媒体内に、物理的に連続した連続記録空間を確保して、該連続記録空間の全体を上記第1のファイルシステムで管理されるようにすると共に、上記連続記録空間内に第2のファイルシステムを配し、上記連続記録空間内でのデータファイルの記録再生が、上記第2のファイルシステムにより管理されるようにしたことを特徴とするファイル管理方法。

【請求項2】 上記第2のファイルシステムは、上記連続記録空間に記録されるデータファイルについての先頭位置情報を管理するとともに、上記連続記録空間に記録されるデータファイルは、物理空間的に連続した領域に所定の規則的処理によるデータ順序で記録されることを特徴とする請求項1に記載のファイル管理方法。

【請求項3】 第1のファイルシステムでデータファイルが管理されるとともに、上記第1のファイルシステムで管理される物理的に連続した連続記録空間内に第2のファイルシステムが配され、上記連続記録空間内でのデータファイルの記録再生が、上記第2のファイルシステムにより管理される記録媒体に対する記録又は再生装置として、上記第1のファイルシステムから上記連続記録空間を判別する判別手段と、

上記連続記録空間において、上記第2のファイルシステムに基づいてデータファイルの記録又は再生のためのアクセスを行うアクセス手段と、を備えたことを特徴とする記録又は再生装置。

【請求項4】 上記連続記録空間へのデータファイルの記録に際しては、上記連続記録空間内の物理空間的に連続した領域に所定の規則的処理によるデータ順序で記録を行うと共に、上記第2のファイルシステムにより、記録したデータファイルについての先頭位置情報を管理させるようにすることを特徴とする請求項3に記載の記録又は再生装置。

【請求項5】 上記第2のファイルシステムは、上記連続記録空間に記録されるデータファイルについての先頭位置情報を管理するとともに、上記連続記録空間に記録されるデータファイルは、物理空間的に連続した領域に所定の規則的処理によるデータ順序で記録されている場合において、

上記第2のファイルシステムからデータファイルの先頭位置情報を判別し、その先頭位置情報で示される位置から物理的に連続した領域からデータを読み出し、上記規則的処理に対応した処理によりデータ再生を行うことを特徴とする請求項3に記載の記録又は再生装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、記録媒体についてのファイル管理方法、及び記録又は再生装置に関するも

のである。

【0002】

【従来の技術】 パーソナルコンピュータやPDA (Personal Digital Assistants: 携帯情報機器) などの情報処理装置においては、HDD (Hard Disc Drive)、光ディスク、光磁気ディスク、磁気ディスク、メモリーカードなどの各種の記録媒体に対するファイル等の記録/再生が行われる。

【0003】

【発明が解決しようとする課題】 ところで一般に、記録媒体に記録されるファイルを管理するためにファイルシステムが構築されるが、その代表的なものとしてFATシステムが知られている。FATについては詳しくは後述するが、このFATは、簡単にいえば記録媒体上のブロックアドレスのリンクを管理することで、複数のブロックで記録されている各ファイルの位置を管理するものである。このFATなどのファイルシステムにより、1つのファイルを記録媒体上で離散的なブロックに記録したり、或いは離散的に記録されたファイルを正しく読み出すことができる。

【0004】 ところで情報処理装置により記録媒体に記録されるファイル内容の種類を考えると、例えばアプリケーションプログラムやデータベースとしてのファイルや、テキストファイル、制御データファイルなどがあり、さらに音楽、音声等のオーディオデータファイル、動画などのビデオデータファイルなど、その内容が時間的に連続性を有するファイルも存在する。なお説明上、オーディオデータファイル、動画ビデオデータファイルなど時間的に連続するファイルをストリームファイルと呼ぶこととする。

【0005】 これらの多様な内容をFATシステムにより記録再生することを考えた場合、FATを参照してブロックリンクをたどって記録再生のアクセスを行うことになるが、上記のストリームファイルの場合は、FATに基づく処理によりシステムの負担が増えることがある。即ち記録時や再生時に時間的な制約のあるストリームファイルの場合は、離散的なリンクをFATのリンクを参照して判別してアクセスしていくことよりは、特にブロック毎にリンクを判別せずに、単純に物理的に連続したブロックにアクセスしていくことができるようにすることが好ましい。

【0006】 また音楽や動画などのストリームファイルの場合は、いわゆる早送り再生や早戻し再生などの要望も発生するが、それらの際にFATから前後のブロックを探していくことも、処理負担が増え、また処理時間も長くなる。なお、早送り、早戻し再生などの場合は、あらかじめストリームファイルを読み出してキャッシュメモリにバッファリングするようにすれば、FATを参照しなくてもよいものとなるが、これには比較的大容量のキャッシュメモリが必要となる。従って例えばPDA機

器などでは実現は困難である。

【0007】さらに音楽データや動画データの場合、著作権の存在するものが多く、記録媒体からの違法なコピーが制限されるようにすることも必要である。

【0008】

【課題を解決するための手段】本発明はこのような状況に鑑みて、例えばストリームデータなどの記録再生に好適なファイル管理方法、及びそれを実現する記録又は再生装置を提供することを目的とする。

【0009】このため本発明のファイル管理方法は、例えばFATなどの第1のファイルシステムでデータファイルが管理される記録媒体内に、物理的に連続した連続記録空間を確保して、該連続記録空間の全体を上記第1のファイルシステムで管理されるようにすると共に、上記連続記録空間内に第2のファイルシステムを配し、上記連続記録空間内でのデータファイルの記録再生が、上記第2のファイルシステムにより管理されるようにする。また上記第2のファイルシステムは、上記連続記録空間に記録されるデータファイルについての先頭位置情報を管理するとともに、上記連続記録空間に記録されるデータファイルは、物理空間的に連続した領域に所定の規則的処理によるデータ順序で記録されるようにするものである。即ち第2のファイルシステムで管理される連続記録空間を確保し、ストリームデータ等の記録再生アクセスをFAT等の第1のファイルシステムによらずに実現できるようにする。

【0010】本発明の、第1のファイルシステムでデータファイルが管理されるとともに、上記第1のファイルシステムで管理される物理的に連続した連続記録空間内に第2のファイルシステムが配され、上記連続記録空間内でのデータファイルの記録再生が、上記第2のファイルシステムにより管理される記録媒体に対する記録又は再生装置は、上記第1のファイルシステムから上記連続記録空間を判別する判別手段と、上記連続記録空間において、上記第2のファイルシステムに基づいてデータファイルの記録又は再生のためのアクセスを行うアクセス手段と、を備えるようにする。また、上記連続記録空間へのデータファイルの記録に際しては、上記連続記録空間内の物理的に連続した領域に所定の規則的処理によるデータ順序で記録を行うと共に、上記第2のファイルシステムにより、記録したデータファイルについての先頭位置情報を管理するようにする。また上記連続記録空間からのデータファイルの再生に際しては、上記第2のファイルシステムからデータファイルの先頭位置情報を判別し、その先頭位置情報で示される位置から物理的に連続した領域からデータを読み出し、上記規則的処理に対応した処理によりデータ再生を行うようにする。

【0011】

【発明の実施の形態】以下、本発明の実施の形態を次の順序で説明する。なお、実施の形態としては、本発明の

ファイル管理方法を実行し、また本発明の記録又は再生装置に相当する、情報処理装置とする。記録媒体としてはメモ리카ードの例を挙げる。

1. 情報処理装置の外観例

2. 情報処理装置の構成

3. OS構造及びデータベース構造

4. メモ리카ード

4-1 外観

4-2 メモ리카ードの端子及び内部構造

4-3 ファイルシステム処理階層

4-4 物理的データ構造

4-5 物理アドレス及び論理アドレスの概念

4-6 論理-物理アドレス変換テーブル

4-7 ディレクトリ構造

5. FAT構造

6. メモ리카ードと情報処理装置のインターフェース

7. メモ리카ードに形成される予約領域（物理空間的連続領域）

8. 予約領域生成処理

9. 予約領域への記録処理

10. 予約領域からの再生処理

【0012】1. 情報処理装置の外観例

本例の情報処理装置の外観例を図1に示す。この情報処理装置1は、いわゆるPDA機器として携帯に適した小型軽量の装置とされる。また記録媒体として、後述するメモ리카ード70を装着し、記録再生を行うことができるものとする。なお本発明としては、携帯型の情報処理装置に限られず、パーソナルコンピュータをはじめとするあらゆるタイプの情報処理装置に適用できるものであり、また装置が記録を行う記録媒体はメモ리카ードに限られず、HDD、光ディスク、光磁気ディスク、或いは装置内に固定的に配置されるRAM、フラッシュメモリなど、他の種の記録媒体であってもよいものである。

【0013】図1(a)(b)(c)(d)は情報処理装置1の外観例としての平面図、右側面図、左側面図、上面図を示している。図1(d)に示すように装置上面側には後述するメモ리카ード70を装着可能なメモリスロット7が形成されており、この情報処理装置1は、メモリスロット7に装着されたメモ리카ード70に対する各種データ（コンピュータ用データ、音楽データ、音声データ、動画データ、静止画像データ、制御データなど）の記録再生が可能とされる。なお、この図1の例ではメモリスロット7が2つ形成されていることから、2つのメモ리카ード70を同時に装着できるようになされている。もちろん、形成するメモリスロット7の数は1つでもよいし、3つ以上でもよい。

【0014】この情報処理装置1には、平面上に例えば液晶パネルによる表示部2が形成され、アプリケーションソフトウェアの起動及び各種処理に伴う画像、データとしての画像や文、再生される音声、音楽に付随する

情報、さらには操作のガイドメッセージ、再生や編集操作等のためのメニュー画面などが表示される。

【0015】情報処理装置1上には、ユーザの操作のための各種の操作子が設けられる。例えば操作キー3a、ジョグダイヤル3b、プッシュダイヤル3cなどがそれぞれ所要部位に形成される。これらの操作子によりユーザは、例えば電源操作、メニュー操作、選択操作、文字等の入力操作、その他必要とされる各種の操作を行うことができる。これらの操作子はもちろん一例にすぎない。即ち配備する操作子の数、種類、位置は多様に考えられる。

【0016】また、情報処理装置1上には、スピーカ4、マイクロホン5、撮像部6も形成され、音声の出力、入力、撮像による画像の取込なども実行できるようにされている。

【0017】また各種機器との接続のために、各種端子が形成される。例えば図1(b)のよう、ヘッドホン端子10、ライン出力端子12、ライン入力端子11などが形成される。また図1(c)のようにIEEE1394端子8、USB(universal serial bus)端子9などが形成される。なお、これらの端子の種類、数、配置位置も、他の例が多様に考えられる。例えば光ケーブル対応のデジタル入出力端子を備えるようにしたり、或いはS CSIコネクタ、シリアルポート、RS232Cコネクタなどが形成されるようにしても良い。

【0018】2. 情報処理装置の構成

図2に情報処理装置1の内部構成を示す。図示するように情報処理装置1内には、まず中核となる部位として、システムコントローラ21、CPU22、フラッシュROM23、D-RAM24が設けられる。また基本的なユーザインターフェースのための部位として操作部35、表示制御部27、表示部2が形成される。

【0019】システムコントローラ21は操作部35からの操作情報を入力し、それに応じてCPU22に割り込みをかける。操作部35とは、図1に示した各種操作子3a、3b、3cに相当する。また図1では説明しなかったが、表示部2に操作キーやアイコンの表示を行うとともに表示部2上でのタッチ検出機構を設けることで、タッチパネル操作子を形成してもよく、その場合のタッチパネル操作子も図2でいう操作部35に含まれるものとなる。

【0020】CPU22は基本ソフト(OS: Operating System)やアプリケーションプログラムが動作される部位となる。CPU22はシステムコントローラ21を介して供給される操作情報に応じて所要の実行する。フラッシュROM23は、基本動作プログラム、各種処理定数、設定情報などを記憶する領域とされる。D-RAM24は、各種処理に必要な情報の記憶、データのバッファリング、CPU22のワークエリアの拡張、その他、CPU22の処理に応じて多様に使用される。

またD-RAM24にはストレージエリア(不揮発性領域)が設けられており、そのストレージエリアにはOSやアプリケーションソフトウェアがインストールされる。そしてD-RAM24にインストールされたアプリケーションソフトウェアは、ユーザからの操作に応じて起動され、CPU22により実行される。またアプリケーションソフトウェアはユーザインターフェース画面を持ち、ユーザの指示による状態遷移に基づいて、D-RAM24に確保されたフレームバッファに描画を行う。描画された画像データは、表示制御部27に送られ、表示部2に表示される。

【0021】また上述したようにメモ리카ード70に対するメモリスロット7が形成され、メモ리카ード70を装着できるが、CPU22は、メモ리카ードインターフェース28を介して装着されたメモ리카ード70に対して書込又は読み出しアクセスすることができる。メモ리카ードインターフェース28とメモ리카ード70との間のインターフェース動作については後述する。CPU22は、装着されたメモ리카ード70を、拡張的なメモリ領域として利用することができる。また、もちろんメモ리카ード70にアプリケーションプログラムが記録されていれば、それをD-RAM24にインストールしたり、或いはアプリケーションデータを直接D-RAM24に展開して所要処理を実行させることができる。また、或るアプリケーションに基づいてCPU22が、作成した文書データ、画像データ、オーディオデータ、表計算データなどを、メモ리카ード70に記録することもできる。なお、メモリスロット7にメモ리카ード70が装着されたことを検出することで、メモ리카ード70に対する動作が記録再生動作可能になったり、或いはメモ리카ード70に記録されているアプリケーションデータが自動的にD-RAM24に展開されるなどの、いわゆるホットプラグイン動作も可能である。またメモ리카ードインターフェース28は、メモ리카ード70に記録するデータについての暗号化処理や、読み出したデータの暗号解読処理なども可能とされる。

【0022】撮像部6は例えばCCD撮像素子及び撮像回路系により形成される。撮像部6により取り込まれた撮像画像データは、撮像データインターフェース34を介してD-RAM24に取り込むことができ、またCPU22は所定のアプリケーションプログラムに基づく動作により、撮像画像データの編集やメモ리카ード70への記録等を実行できる。

【0023】オーディオインターフェース29は、上述したスピーカ4、マイクロホン5、ヘッドホン端子10、ライン出力端子12、ライン入力端子11から入出力されるオーディオデータのインターフェース部位となる。例えばマイクロホン5或いはライン入力端子11から入力されたアナログオーディオ信号は、入力オーディオ処理部32でそれぞれ所定の増幅処理やフィルタリ

グが行われ、A/D変換器33でデジタルオーディオデータとされてオーディオインターフェース29に供給される。オーディオインターフェース29は、入力されたデジタルオーディオデータについて、CPU22の制御に基づいて処理や出力を実行する。例えば所要の圧縮エンコード処理を行った後、メモリカードインターフェース28に供給し、メモリカード70に記録させることができる。またオーディオインターフェース29は、例えばメモリカード70から読み出されるなどして供給されたデジタルオーディオデータについて所定のデコード処理を行い、D/A変換器30に供給する。D/A変換器30はデジタルオーディオデータをアナログオーディオ信号に変換する。出力オーディオ処理部31は供給されたアナログオーディオ信号として、出力先に応じた所定の増幅処理、インピーダンス調整などを行い、スピーカ4、ヘッドホン端子10、ライン出力端子12に出力する。

【0024】USBインターフェース25は、USBコネクタ9に接続された外部機器との間の通信インターフェースである。CPU22はUSBインターフェース25を介して外部のパーソナルコンピュータ或いは周辺機器などとの間でデータ通信を行うことができる。例えばこの情報処理装置1で扱われる制御データ、コンピュータデータ、画像データ、オーディオデータなどの送受信が実行される。同様にIEEE1394インターフェース26は、IEEE1394端子8に接続された外部機器との間の通信インターフェースである。CPU22はIEEE1394インターフェース26を介して外部の情報機器との間で各種データ通信を行うことができる。

【0025】なお、この図2に示す情報処理装置1の構成はあくまでも一例であり、これに限定されるものではない。即ち、一般にパーソナルコンピュータやPDA機器で採用されている各種構成部位を追加したり、或いは実際の製品として不要の部位を削除することは、設計上の都合により決められるものである。

【0026】3. OS構造及びデータベース構造
続いて図3で、本例の情報処理装置1に記載されるOS構造について説明する。図3に示すように、OSは、基本ソフトの中心部分としてのカーネルを含むマネージャ層と、標準ライブラリ、及び制御ICなどのハードウェアのレイヤとなるHAL (Hardware Abstraction Layer) から構成される。アプリケーションソフトウェアは、このようなOS構造による基本動作上で動作される。またHALに対しては、1又は複数のデバイスドライバとして階層が付加され実際のハードウェア (HW) が駆動される。

【0027】ここで、特に本例の情報処理装置1の場合には、メモリカード70をドライブ可能とし、かつ後述するメモリカード70のデータはFATにより管理されることから、OSにFATライブラリが付加され、さら

に、メモリカードをハンドリングするためのライブラリ (MSライブラリ) が付加される。そしてこのFATライブラリ及びMSライブラリに基づいて、メモリドライブがメモリカード70がドライブされる構造とされている。

【0028】このようなOS構造を持つ本例の情報処理装置1では、さらに通常でいうところの「ファイル」に相当する概念として、「データベース」という概念が導入されている。ここでいう「データベース」とは、通常いうところのデータベースのように単にデータを蓄積していったものではなく、データベース自体がデータを管理できる構造としてフォーマット化されている。この意味で、「データベース」は「ファイル」に相当する。

【0029】図4にデータベース構造を示す。即ちデータベースには、ヘッダ (DTBヘッダ) としてデータベースネーム (DTB Name) 及びその他情報を含む領域が形成され、さらにポインタテーブルが配される。そしてデータ領域に記録される実際のデータは、ポインタテーブルに記録されたポインタ情報により、位置的な管理が行われる状態となっている。

【0030】このような構造のデータベースとしては、2種類のものが存在する。例えば一般に1つのアプリケーションソフトウェアは複数のファイルで構成され、その中には実行ファイル (*.exe) と、データファイル (*.data) があるが、その実行ファイル (*.exe) に相当するものとして「リソースデータベース (*.prc)」があり、またデータファイル (*.data) に相当するものとして「データベースデータベース (*.dtb)」がある。

【0031】本例の情報処理装置1では、このような「データベース」という概念によりデータを扱う。従って、メモリカード70において記録再生されるファイル (FATで扱われるファイル) も、上記データベースの形態となる。なお本明細書では、「ファイル」という言葉を用いるが、これは一般的な概念にあわせて用いているものであり、本実施の形態に例示して言えば、「ファイル」とは上記構造のデータベースの意味となる。

【0032】4. メモリカード

4-1 外観

次にメモリカード70について説明していく。まず図5にメモリカード70の外形状を示す。メモリカード70は、例えば図5に示すような板状の筐体内部に例えば所定容量のメモリ素子を備える。本例としては、このメモリ素子としてフラッシュメモリ (Flash Memory) が用いられるものである。図5に平面図、正面図、側面図、底面図として示す筐体は例えばプラスチックモールドにより形成され、サイズの実例としては、図1に示す幅W11、W12、W13のそれぞれが、W11=60mm、W12=20mm、W13=2.8mmとなる。

【0033】筐体の正面下部から底面側にかけて例えば10個の電極を持つ端子部72が形成されており、この端子部72から、内部のメモリ素子に対する読出又は書込動作が行われる。筐体の平面方向の左上は切欠部73とされる。この切欠部73は、このメモリカード70を、例えばドライブ装置本体側の着脱機構へ装填する際に挿入方向を誤ることを防止するためのものとなる。また筐体上面から底面側にかけて、ラベル貼付面74が形成され、ユーザーが記憶内容を書いたラベルを貼付できるようにされている。さらに底面側には、記録内容の誤消去を防止する目的のスライドスイッチ75が形成されている。

【0034】このようなメモリカード70においては、フラッシュメモリ容量としては、4MB（メガバイト）、8MB、16MB、32MB、64MB、128MBの何れかであるものとして規定されている。またデータ記録/再生のためのファイルシステムとして、いわゆるFAT（File Allocation Table）システムが用いられている。

【0035】書込速度は1500KByte/sec～330KByte/sec、読出速度は2.45MByte/secとされ、書込単位は512バイト、消去ブロックサイズは8KB又は16KBとされる。また電源電圧Vccは2.7～3.6V、シリアルクロックCLKは最高20MHzとされる。

【0036】4-2 メモリカードの端子及び内部構造
図6に端子部72の電極構造を示す。図5に示したように端子部72は10個の平面電極が1列に並んだ構造とされるが、図6に示すように各電極（端子T1～T10）は次の通りとなる。

【0037】端子T1及びT10は検出電圧Vss端子とされる。端子T2は、シリアルプロトコルバスステータス信号BSの入力端子とされる。端子T3及びT9は電源電圧Vcc端子とされる。端子T4はデータ端子、つまりシリアルプロトコルデータ信号の出力端子とされる。端子T5及びT7はリザーブ（予備）とされる。端子T6は検出端子とされ、ドライブ装置側（情報処理装置1のメモリカードインターフェース）がメモリカードの装着検出に用いる。端子T8は、シリアルクロックSCLKの入力端子とされる。

【0038】また図6にはメモリカード70の内部構成も示している。メモリカード70の内部は、コントロールIC80とフラッシュメモリ81が設けられている。コントロールIC80はフラッシュメモリ81に対する書込/読出動作を実行する部位となる。図からわかるように、コントロールIC80に対しては、端子T2からのシリアルプロトコルバスステータス信号BS、端子T8からのシリアルクロックSCLKが供給される。書込動作時には、コントロールIC80は、これらのシリアルプロトコルバスステータス信号BS、シリアルクロック

SCLKに従って、端子T4から供給されるデータのフラッシュメモリ81への書込を行う。また読出時には、シリアルプロトコルバスステータス信号BS、シリアルクロックSCLKに従って、フラッシュメモリ81からデータを読み出し、端子T4からドライブ装置側に出力する。

【0039】また検出電圧Vssは、検出端子T6に供給されており、ドライブ装置側では、図示のように抵抗Rによって検出端子T6の端子電圧を検出することで、このメモリカード70が装着部（メモリスロット7）に接続されているか否かを検出できるようにされる。

【0040】4-3 ファイルシステム処理階層
続いて、メモリカード70を記録媒体とするシステムにおけるフォーマットについて説明していく。図7は、メモリカード70を記録媒体とするシステムのファイルシステム処理階層を示すものである。この図に示すように、ファイルシステム処理階層としては、アプリケーション処理層の下に、順次、ファイル管理処理層、論理アドレス層、物理アドレス層、フラッシュメモリアクセスがおかれる。この階層では、ファイル管理処理層がいわゆるFAT（File Allocation Table）となる。また、この図から分かるように、本例のファイルシステムでは論理アドレス及び物理アドレスという概念が導入されているが、これについては後述する。

【0041】4-4 物理的データ構造
図8には、メモリカード70内の記憶素子である、フラッシュメモリ81の物理的データ構造が示されている。フラッシュメモリ81としての記憶領域は、セグメントという固定長のデータ単位が元となる。このセグメントは、1セグメントあたり4MB（メガバイト）或いは8MBとして規定されるサイズであり、1つのフラッシュメモリ81内におけるセグメント数は、そのフラッシュメモリ81の容量に依存して異なってくる。

【0042】そして、この1セグメントを図8（a）に示すように、ブロックという固定長のデータ単位として8KB（キロバイト）又は16KBにより区切るようにされる。原則として、1セグメントは512ブロックに区切られることから、図8（a）に示すブロックnについては、 $n=511$ とされることになる。但し、フラッシュメモリ81では、書き込み不可な損傷エリアであるディフェクトエリアとしてのブロック数が所定数の範囲で許可されているため、データ書き込みが有効とされる実質的なブロック数を対象とすれば、上記nは511よりも少なくなる。

【0043】図8（a）に示すようにして形成されるブロック0～nのうち、先頭の2つのブロック0、1はブートブロックといわれる。但し、実際には有効なブロックの先頭から2つのブロックがブートブロックとして規定されることになっており、必ずしもブートブロックが

ブロック0, 1である保証はない。そして、残りのブロックが、ユーザデータが格納されるユーザブロックとなる。

【0044】1ブロックは、図8(d)に示すようにして、ページ0~mにより分割される。1ページの容量は、図8(e)に示すように、512バイトのデータエリアと16バイトの冗長部よりなる。528(=512+16)バイトの固定長とされる。なお、冗長部の構造については図8(f)により後述する。また、1ブロック内のページ数としては、1ブロックの容量が8KBの場合には16ページ、16KBの場合には32ページとなる。

【0045】このような、図8(d)(e)に示されるブロック内のページ構造は、上記ブートブロックとユーザブロックとで共通である。また、フラッシュメモリ81では、データの読み出し、及び書き込みはページ単位で行われ、データの消去はブロック単位で行われるものとされる。そして、データの書き込みは、消去済みのページに対してしか行われないものとされている。従って、実際のデータの書き換えや書き込みは、ブロック単位を対象として行われることになる。

【0046】先頭のブートブロックは、図8(b)に示すように、ページ0に対してヘッダが格納され、ページ1には初期不良データの位置(アドレス)を示す情報が格納される。また、ページ2にはCIS/IDIといわれる情報が格納される。2つめのブートブロックは図8(c)に示すように、ブートブロックとしてのバックアップのための領域とされている。

【0047】図8(e)に示された冗長部(16バイト)は、図8(f)に示す構造を有する。この冗長部は、図のように先頭の第0バイト~第2バイトの3バイトが、データエリアのデータ内容の更新に応じて書き換えが可能なオーバーライトエリアとされる。このオーバーライトエリアのうち、第0バイトにはブロックステータスが格納され、第1バイトにはデータステータスが格納される(Block Flag Data)。また、第2バイトの上位の所定ビットを利用して交換テーブルフラグ(Page Data Status1)が格納される。

【0048】原則として第3バイト~第15バイトは、その内容が現ページのデータ内容に応じて固定とされ、書き換えが不可とされる情報が格納される領域となる。第3バイトにはアクセス許可やコピー禁止指定等を示す管理フラグ(Block Info)が格納される。第4、第5バイトから成る2バイトの領域には、後述する論理アドレス(Logical Address)が格納される。第6~第10バイトの5バイトの領域は、フォーマットリザーブの領域とされ、続く第11、第12バイトの2バイトは、上記フォーマットリザーブに対して誤り訂正を施すための分散情報ECCを格納する領域とされる。残る第13~第15バイトには、図8(e)に示すデータエリアのデータに対し

て誤り訂正を行うためのデータECCが格納される。

【0049】上記図8(f)に示した冗長部の第3バイトに格納される管理フラグは、図9に示すようにして、ビット7~ビット0の各ビットに、その内容が定義されている。ビット7、6、及びビット1、0はリザーブ(未定義)領域とされている。ビット5は現ブロックに対してのアクセス許可の「有効」('1'; Free) / 「無効」('0'; Read Protected)を示すフラグが格納される。ビット4には現ブロックについてのコピー禁止指定('1'; OK / '0'; NG)についてのフラグが格納される。

【0050】ビット3は交換テーブルフラグとされる。この交換テーブルフラグは、現ブロックが後述する論理-物理アドレス交換テーブルであるのか否かを示す識別子であり、このビット3の値が「0」とされていれば、現ブロックは論理-物理アドレス交換テーブルであることが識別され、「0」であれば無効となる。つまり、現ブロックは論理-物理アドレス交換テーブルではないことが識別される。

【0051】ビット2はシステムフラグが格納され、「1」であれば現ブロックがユーザブロックであることが示され、「0」であればブートブロックであることが示される。

【0052】ここで、セグメント及びブロックと、フラッシュメモリ容量との関係を図13(左列を参照)により説明しておく。メモリアドレス0のフラッシュメモリ容量としては、4MB、8MB、16MB、32MB、64MB、128MBの何れかであるものとして規定されている。そして、最も容量の小さい4MBの場合であると、1ブロックは8KBと規定された上で、そのブロック数としては512個とされる。つまり、4MBはちょうど1セグメントの容量を有するものとされる。そして、4MBの容量であれば、同様に1ブロック=8KBの容量が規定された上で、2セグメント=1024ブロックとなる。なお、前述したように、1ブロック=8KBであれば、1ブロック内のページ数は16ページとなる。但し16MBの容量では、1ブロックあたりの容量として8KBと16KBの両者が存在することが許可されている。このため、2048ブロック=4セグメント(1ブロック=8KB)のもの、1024ブロック=2セグメント(1ブロック=16KB)のものとの2種類があることになる。1ブロック=16KBの場合には、1ブロック内のページ数は32ページとなる。

【0053】また、32MB、64MB、128MBの容量では、1ブロックあたりの容量は16KBのみであるとして規定される。従って、32MBでは2048ブロック=4セグメントとなり、64MBでは4096ブロック=8セグメントとなり、128MBでは8192ブロック=16セグメントとなる。

【0054】4-5 物理アドレス及び論理アドレスの

概念

次に、上述したようなフラッシュメモリの物理的データ構造を踏まえたうえで、図10に示すデータ書き換え動作に従って、本例のファイルシステムにおける物理アドレスと論理アドレスの概念について説明する。

【0055】図10(a)には、或るセグメント内から4つのブロックを抜き出して、これを模式的に示している。各ブロックに対しては物理アドレスが付される。この物理アドレスはメモリにおけるブロックの物理的な配列順に従って決まるもので、或るブロックとこれに対応付けられた物理アドレスとの関係は不変となる。ここでは、図10(a)に示す4ブロックに対して、上から順に物理アドレスの値として、105、106、107、108が付されている。なお、実際の物理アドレスは2バイトにより表現される。

【0056】ここで、図10(a)に示すように、物理アドレス105、106で示されるブロックがデータの記憶されている使用ブロックで、物理アドレス107、108で示されるブロックがデータが消去（即ち、未記録領域）された未使用ブロックとなっている状態であるとする。

【0057】そして、論理アドレスであるが、この論理アドレスは、ブロックに対して書き込まれたデータに付随するように割り振られるアドレスとされる。そして、この論理アドレスが、後述するFATファイルシステムが利用するアドレスとされている。図10(a)では、4つの各ブロックに対して、上から順に論理アドレスの値として、102、103、104、105が付されている状態が示されている。なお、論理アドレスも実際には2バイトにより表現されるものがある。

【0058】ここで、上記図10(a)に示す状態から、例えば物理アドレス105に格納されているデータの更新として、内容の書き換え又は一部消去を行うとする。このような場合、フラッシュメモリのファイルシステムでは、同じブロックに対して更新したデータを再度書き込むことはせずに、未使用のブロックに対してその更新したデータを書き込むようにされる。つまり、例えば図10(b)に示すようにして、物理アドレス105のデータは消去したうえで、更新されたデータはこれまで未使用ブロックであった物理アドレス107で示されるブロックに書き込むようにされる（処理①）。

【0059】そして、処理①として示すように、データ更新前（図10(a)）の状態では物理アドレス105に対応していた論理アドレス102が、更新されたデータが書き込まれたブロックの物理アドレス107に対応するように、論理アドレスについての変更を行うものである。これに伴って、データ更新前は物理アドレス107に対応していた論理アドレス104については、物理アドレス105に対応するように変更されている。

【0060】つまり、物理アドレスはブロックに対して

固有に付されるアドレスであり、論理アドレスは、一旦ブロックに対して書き込まれたデータに付随するようにしてついて回る。ブロック単位の書き込みデータに固有となるアドレスであることができる。

【0061】このようなブロックのスワップ処理が行われることで、或る同一の記憶領域（ブロック）に対して繰り返し集中的にアクセスされることが無くなり、書き換え回数の上限があるフラッシュメモリの寿命を延ばすことが可能となる。そして、この際に論理アドレスを上記処理①のようにして扱うことで、ブロックのスワップ処理によって更新前と更新後のデータとで書き込まれるブロックの移動があるようにされても、FATからは同一のアドレスが見えることになり、以降のアクセスを適正に実行することができるものである。なお、後述する論理-物理アドレス交換テーブル上での更新のための管理を簡略にするなどを目的として、ブロックのスワップ処理は、1セグメント内で完結するものとして規定されている。逆に言えば、ブロックのスワップ処理はセグメント間で跨るようには行われない。

【0062】4-6 論理-物理アドレス交換テーブル
上記図10による説明から分かるように、ブロックのスワップ処理が行われることで、物理アドレスと論理アドレスの対応は変化する。従って、フラッシュメモリに対するデータの書き込み及び読み出しのためのアクセスを実現するには、物理アドレスと論理アドレスとの対応が示される論理-物理アドレス交換テーブルが必要となる。つまり、論理-物理アドレス交換テーブルをFATに参照することで、FATが指定した論理アドレスに対応する物理アドレスが特定され、この特定された物理アドレスにより示されるブロックにアクセスすることが可能になるものである。逆に言えば、論理-物理アドレス交換テーブルが無ければ、FATによるフラッシュメモリへのアクセスが不可能となる。

【0063】従来では、例えばセット本体に対してメモリカード70が装着されたときに、セット本体側のマイクロプロセッサがメモリカード70の記憶内容をチェックすることで、セット本体側で論理-物理アドレス交換テーブルの構築を行い、更にこの構築された論理-物理アドレス交換テーブルをセット本体側のRAMに格納するようにしていた。つまり、メモリカード70内には、論理-物理アドレス交換テーブルの情報は格納されてはなかった。これに対して本例では、以降説明するようにメモリカード70に対して、論理-物理アドレス交換テーブルを格納するように構成している。

【0064】図11は、本例のメモリカード70に対して格納される論理-物理アドレス交換テーブルの構築形態を概念的に示すものである。つまり、本例では、例えば論理アドレスの昇順に従って、これに対応する2バイトの物理アドレスを格納するようにしたテーブル情報を論理-物理アドレス交換テーブルとして構築するように

される。なお、前述したように、物理アドレス、及び論理アドレスは共に2バイトで表現される。これは、128MBの最大容量のフラッシュメモリの場合には8192個のブロックが存在する。最大で、この8192個のブロック数をカバーできるだけのビット数が必要とされることに基づく。このため、図11において例示している物理アドレスと論理アドレスについても、実際に即して2バイトで表現している。但し、ここでは、この2バイトを16進数により表記している。つまり、「0x」によりその後続く値が16進法表記であることが示される。なお、この「0x」により16進数であることを表す表記は、以降の説明において16進数を表記する場合にも同様に用いることとする。(但し、表記の煩雑化を防ぐため「0x」を省略している図面もある。)

【0065】図12に、上記図11に示した概念に基づく論理-物理アドレス交換テーブルの構造例を示す。論理-物理アドレス交換テーブルは、フラッシュメモリの最後のセグメント内の或るブロックに対して、図12に示すようにして格納される。先ず図12(a)に示すように、ブロックを分割するページのうち、ページ0、1からなる2ページの領域がセグメント0用の論理-物理アドレス交換テーブルとして割り当てられる。例えば、図13にて説明したように、フラッシュメモリが4MBの容量であれば1セグメントしか有さないために、このページ0、1のみの領域が論理-物理アドレス交換テーブルの領域となる。また、例えばフラッシュメモリが8MBの容量であれば2セグメントを有するため、セグメント0用の論理-物理アドレス交換テーブルとして割り当てられるページ0、1に加え、これに続くページ2、3の2ページがセグメント1用の論理-物理アドレス交換テーブルとして割り当てられることになる。

【0066】以降、フラッシュメモリの容量の増加に応じて、続く2ページごとにセグメントごとの論理-物理アドレス交換テーブルの割り当て領域が設定されていくことになる。そして、最大の128MBの容量を有する場合であれば16セグメントが存在するため、最大では、セグメント15用までのページが論理-物理アドレス交換テーブルの領域として割り当てられることになる。従って、最大の128MBの容量のフラッシュメモリでは、30ページが使用されることになり、図12(a)に示すページNとしては、最大でN=29となる。これまでの説明から分かるように、論理-物理アドレス交換テーブルは、セグメントごとに管理されるものである。

【0067】図12(b)は、1セグメントあたりの論理-物理アドレス交換テーブルの構造を示すものとして、2ページ分のデータエリアを抜き出して示している。つまり、1ページのデータエリアは512バイト(図8(e)参照)であることから、図12(b)に

は、1024(=512×2)バイトが展開されている状態が示されている。

【0068】図12(b)に示すように、この2ページ分のデータエリアである1024バイトについて2バイトごとに区切り、この2バイトごとの領域を、先頭から順次、論理アドレス0用、論理アドレス1用……、のようにして割付を行い、最後は先頭から991バイト目と992バイト目の2バイトの領域を論理アドレス495用の領域として割り付けるように規定を行う。これら2バイトごとの領域に対して、各論理アドレスが対応する物理アドレスを書き込むようにする。従って、本例の論理-物理アドレス交換テーブルでは、実際のデータ更新によるブロックのスワップ処理などにより物理アドレスと論理アドレスの対応が変更された場合には、論理アドレスを基準として、物理アドレスの格納状態が更新されるようにしてテーブル情報の書き換えが行われることになる。

【0069】また、残る993バイト目から最後の1024バイト目までの計32バイトの領域は、余剰ブロックの物理アドレスが格納される領域として割り当てられる。つまり、16個の余剰ブロックの物理アドレスを管理することができる。ここでいう余剰ブロックとは、例えばブロック単位でデータの更新を行う際に書き換え対象となるデータを一時待避させる領域として設定されたいわゆるワークブロックなどを言うものである。

【0070】ところで、1セグメントは512ブロックに分割されているものであると先に説明したのにも関わらず、図12に示したテーブル構造では、管理可能なブロック数が論理アドレス0用-論理アドレス495用の496ブロックとしている。これは、実際上、上記した余剰アドレスが設定されることと、前述したように、フラッシュメモリでは、或ブロック数のディフェクト(使用不可領域)が許可されている。そのため現実には、相当数のディフェクトブロックが存在することに依る。従って、実際には、書き込み/消去が有効なブロックを管理するのに、496ブロックを管理できるように構成しておけば充分とされるものである。

【0071】そして、このようにして論理-物理アドレス交換テーブルが格納されるブロックについては、これを形成する各ページの冗長部における管理フラグ(図9参照)のデータ内容として、この管理フラグのビット3に対して'0'がセットされることになる。これにより、当該ブロックが論理-物理アドレス交換テーブルが格納されているブロックであることが示されることになる。

【0072】論理-物理アドレス交換テーブルが格納されるブロックも、論理-物理アドレス交換テーブルの内容の書き換えがあった場合には、例外なく、先に図10にて説明したスワップ処理が行われる。従って、論理-物理アドレス交換テーブルが記録されているブロックは

不定であり、或る特定のブロックに論理-物理アドレス変換テーブルを格納するように規定することは出来ない。そこで、FATは、フラッシュメモリにアクセスして上記した管理フラグのビット3が「0」とされているブロックを検索することで、論理-物理アドレス変換テーブルが格納されているブロックを識別するようにされる。但し、論理-物理アドレス変換テーブルが格納されているブロックの検索がFATによって容易に行われるようにすることを考慮して、論理-物理アドレス変換テーブルが格納されているブロックは、そのフラッシュメモリにおける最後のナンバが付されたセグメントに在るように、本例では規定するものとされる。これにより、FATは最後のナンバが付されたセグメントのブロックのサテライトで、論理-物理アドレス変換テーブルを検索することができる。つまり、論理-物理アドレス変換テーブルを検索するのに、フラッシュメモリの全てのセグメントを検索するの必要は無いようにされる。上記図12に示した論理-物理アドレス変換テーブルは、例えばメモ리카ード70の製造時において格納するようにされる。

【0073】ここで、再度図13を参照して、フラッシュメモリ容量と論理-物理アドレス変換テーブルのサイズとの関係を説明しておく。上記図11にて説明したように、1セグメントを管理するための論理-物理アドレス変換テーブルのサイズは2ページ分の1024バイト、つまり1KBとなる。従って、図13の最右列に記されているように、フラッシュメモリが4MB(1セグメント)の容量では論理-物理アドレス変換テーブルは1KBのサイズとなる。また、フラッシュメモリの容量が8MB(2セグメント)では論理-物理アドレス変換テーブルは2KB(4ページ)となる。また、フラッシュメモリの容量が16MBの場合、2048ブロック=4セグメントのもので論理-物理アドレス変換テーブルは4KB(8ページ)、1024ブロック=2セグメントのもので論理-物理アドレス変換テーブルは2KB(4ページ)となる。そして、フラッシュメモリの容量が32MB(4セグメント)では論理-物理アドレス変換テーブルは4KB(8ページ)、フラッシュメモリの容量が64MB(8セグメント)では論理-物理アドレス変換テーブルは8KB(16ページ)となり、フラッシュメモリの容量が最大の128MB(16セグメント)では論理-物理アドレス変換テーブルは16KB(32ページ)となる。

【0074】4-7 ディレクトリ構造

メモ리카ード70に記録されるディレクトリ構成例を図14に示す。メモ리카ード70で扱うことのできる主データとしては、コンピュータ用データ、動画データ、静止画データ、メッセージデータ、オーディオデータ、制御用データなどがあるが、このためディレクトリ構造としては、ルートディレクトリから、「VOICE」(メ

ッセージ用ディレクトリ)、「DCIM」(静止画用ディレクトリ)、「MOxxxxxx」(動画用ディレクトリ)、「CONTROL」(制御用ディレクトリ)、「HIFI」(オーディオ用ディレクトリ)、「PM」(情報処理装置用ディレクトリ)が配される。

【0075】そして図示していないが、各ディレクトリの下には、サブディレクトリやファイル(上述したデータベース)、フォルダ等が配され、いわゆるツリー構造の形態をとることになる。なお、もちろんこのようなディレクトリ構成は一例にすぎず、実際には情報処理装置1等による記録状況や記録されるファイル種別などに応じてディレクトリ構造が形成される。

【0076】5. FAT構造

図7のファイルシステム階層で説明したように、ファイル管理処理はFATにより行われることになる。即ち図2に示した構成の情報処理装置1により、メモ리카ード70に対する記録再生(データ書込/読出)を実現するには、アプリケーション処理での要求に伴ってFATによるファイル記憶位置管理が参照され、さらに上述した論理-物理アドレス変換が行われて実際のアクセスが行われることになる。ここで、FATの構造について説明しておく。

【0077】図15はFATによる管理構造の概要を示している。なお、本例ではFAT及び論理-物理アドレス変換テーブルはメモ리카ード70内に格納されることになるが、図15に示すFAT構造が、メモ리카ード70内での管理構造となるものである。

【0078】図示するようにFAT管理構造は、パーティションテーブル、空き領域、ブートセクタ、FAT、FATのコピー、ルートディレクトリ、データ領域から成る。データ領域には、クラスタ2、クラスタ3...として単位データを示しているが、このクラスタとは、管理単位となるFATで扱う1データ単位である。一般にFATでは、クラスタサイズは標準で4KBとされるが、このクラスタサイズは512バイト~32KBバイトの間で2のべき乗の大ききとすることができる。本例のメモ리카ード70では、上述したように1つのブロックが8KBバイト又は16KBバイトとされるが、1ブロック=8KBバイトとされるメモ리카ード70の場合は、FATで扱うクラスタは8KBバイトとされる。また1ブロック=16KBバイトとされるメモ리카ード70の場合は、FATで扱うクラスタは16KBバイトとされる。即ち、8KBバイト又は16KBバイトがFAT管理上のデータ単位であり、かつメモ리카ード70でのブロックとしての1つのデータ単位とされる。なお、従ってメモ리카ードからみれば、FATで扱われるクラスタサイズ=そのメモ리카ードのブロックサイズとなる。このため、本例の以降の説明については、簡略化のためにブロック=クラスタとして考えることとする。

【0079】そして図15左側にブロックナンバとして

$x \cdots (x+m-1), (x+m), (x+m+1), (x+m+2) \cdots$ と示したが、例えばこのように各ブロックにおいてFAT構造を構築する各種データは記憶されることになる。なお、実際には必ずしもこのように物理的に連続する各ブロックに各情報が記憶されるものではない。

【0080】FAT構造において、まずパーティションテーブルには、FATパーティション（最大2GBバイト）の先頭と終端のアドレスが記述されている。ブート領域には、いわゆる12bitFAT、16bitFATの別や、FAT構造（大きさ、クラスタサイズ、各領域のサイズなど）が記述される。

【0081】FATは、後述するように各ファイルを構成するクラスタのリンク構造を示すテーブルとなり、またFATについては続く領域にコピーが記述される。ルートディレクトリには、ファイル名、先頭クラスタ番号、各種属性が記述される。これらの記述は1つのファイルにつき32バイト使用される。

【0082】FATにおいては、FATのエントリとクラスタは1対1で対応しており、各クラスタのエントリにはリンク先、つまり後に続くクラスタの番号が記述される。つまり、複数のクラスタ（＝ブロック）で形成されている或るファイルについてみると、まずディレクトリによって先頭のクラスタ番号が示され、FATにおけるその先頭クラスタのエントリには、次のクラスタ番号が示される。さらに次のクラスタ番号のエントリには、さらに次のクラスタ番号が示される。このようにクラスタのリンクがFATに記述される。

【0083】図16はこのようなリンクの概念を模式的に示している（数値は16進値）。例えば2つのファイル「MAIN.C」「FUNC.C」が存在するとすると、ディレクトリにはこの2つのファイルの先頭クラスタ番号が例えば「002」「004」と記述される。そしてファイル「MAIN.C」については、クラスタ番号「002」のエントリに次のクラスタ番号「003」が記述され、またクラスタ番号「003」のエントリに次のクラスタ番号「006」が記述される。さらに、クラスタ番号006がこのファイル「MAIN.C」の最後のクラスタであるとして、クラスタ番号「006」のエントリには、最後のクラスタであることを示す「FFF」が記述される。これによりファイル「MAIN.C」がクラスタ「002」→「003」→「006」という順番で記憶されている。即ち、仮にクラスタ番号とメモリアドレス70でのブロック番号が一致していると仮定すると、ファイル「MAIN.C」は、メモリアドレス70内でブロック「002」「003」「006」に記憶されていることが表現されている。（但し、FATで扱うクラスタは、上述のように論理アドレスで扱うものとなるため、ブロックの物理アドレスとそのまま一致するものではない）

【0084】また同様にファイル「FUNC.C」については、FATにより、クラスタ「004」→「005」に記憶されていることが表現される。

【0085】なお、未使用のブロックに対応するクラスタについては、そのエントリは「000」とされる。

【0086】ところでルートディレクトリの領域に記憶される各ファイルのディレクトリにおいては、図16に示した先頭クラスタ番号だけでなく、例えば図17のように各種データが記述される。即ちファイル名、拡張子、属性、変更時刻情報、変更日付情報、先頭クラスタ番号、ファイルサイズが、それぞれ図示するバイト数で記述される。

【0087】また或るディレクトリの下層となるサブディレクトリについては、図15のルートディレクトリの領域ではなく、データ領域に記憶される。つまりサブディレクトリは、ディレクトリ構造を持つファイルとして扱われる。そしてサブディレクトリの場合はサイズは無制限とされ、また自分自身へのエントリと親ディレクトリへのエントリが必要になる。

【0088】図18に、或るルートディレクトリ内にファイル「DIR1」（属性＝ディレクトリ；つまりサブディレクトリ）があり、さらにその中にファイル「DIR2」（属性＝ディレクトリ；つまりサブディレクトリ）があり、さらにその中にファイル「FILE」が存在する場合の構造例を示している。つまりルートディレクトリの領域には、サブディレクトリであるファイル「DIR1」としての先頭クラスタ番号が示され、上述したFATにより、クラスタX、Y、Zがリンクされている状態となる。この図からわかるように、サブディレクトリ「DIR1」「DIR2」についてはファイルとして扱われてFATのリンクに組み込まれる。

【0089】6. メモリカードと情報処理装置のインターフェース

図19により、メモリアドレス70と情報処理装置1のメモリアドレスインターフェース28の間のシリアルインターフェースシステム構成を説明する。メモリアドレス70内のコントロールIC80は、図19に示すようにフラッシュメモリコントローラ80a、レジスタ80b、ページバッファ80c、シリアルインターフェース80dとして各ブロックを有するものとなっている。

【0090】フラッシュメモリコントローラ80aは、レジスタ80bに設定されたパラメータに基づいて、フラッシュメモリ81とページバッファ80cの間でデータ転送を行う。そしてページバッファ80cにバッファリングされたデータはシリアルインターフェース80dを介して情報処理装置1のメモリアドレスインターフェース28側に転送され、また情報処理装置1のメモリアドレスインターフェース28から転送されてきたデータはシリアルインターフェース80dを介してページバッファ80cにバッファリングされる。

【0091】メモリアードインターフェース28側では、メモリアード70に対するインターフェース構造として、ファイルマネージャ60、転送プロトコルインターフェース61、シリアルインターフェース62を有する。ファイルマネージャ60はメモリアード70のファイル管理を行う。例えば本例のシステムではメモリアード70内にはメインデータファイルの管理のための管理ファイルが記憶されているが、情報処理装置1は装填されたメモリアード70から管理ファイルを読みこんでCPU22がファイルマネージャ60を形成することになる。メモリアード70へのアクセスはファイルマネージャ60に従って実行される。転送プロトコルインターフェース61は、レジスタ80b、ページバッファ80cへのアクセスを実行する。シリアルインターフェース62は、メモリアード70との間の3つの信号線、即ちSCLK（シリアルクロック）、BS（バーステイト）、SDIO（シリアルデータ入出力）において、任意のデータ転送を行うためのプロトコルを規定する。

【0092】以上の構成における各部の動作により、情報処理装置1によるメモリアード70（フラッシュメモリー81）に対する読み出しアクセス/書き込みアクセスが実行される。

【0093】7. メモリアードに形成される予約領域（物理空間的連続領域）

続いて本例の情報処理装置1による特徴的な動作となるメモリアード70に対するファイル管理方法について説明していく。

【0094】図20は本例のファイル管理形態を模式的に示したものである。上述したようにメモリアード70においては、基本的にFATシステムによりファイル管理が行われるが、例えば図20では、ファイルFL1、FL2、FL3が記録された状態を示している。なお、メモリアード70に記録されるファイルとは、上述したリソースデータベース（**、prc）やデータベース（**、dtb）に相当するものである。図20のファイルFL1、FL2、FL3は、FATシステムにより管理されるものであり、例えばファイルFL1は、ファイル構成部分FL1（1）、FL1（2）、FL1（3）から形成されるが、矢印で示すように、これらの記録されたブロックのリンクがFATで管理される。ファイルFL2、FL3も同様にFATで管理されている。

【0095】ここで本例の場合は、メモリアード70上の記録空間において、予約領域FL-Rとして示すように、物理的に連続する記録空間（連続記録空間）を確保することができるようにしている。例えば比較的大容量の空間として予約領域FL-Rを確保する。そして、この予約領域FL-Rの全体、即ち予約領域FL-Rを構成する複数の物理空間的に連続したブロックは、FATからは1つの巨大ファイルとして管理されることにな

る。ただしFATからは、あくまで予約領域FL-Rの全体が1つの仮想的なファイルとして見えるのみとされ、この予約領域FL-R内に記録される個々のファイルは管理されない。つまりFATからは予約領域FL-R全体の位置がわかるのみである。

【0096】予約領域FL-R内の管理は、この予約領域FL-Rに配される第2のファイルシステムによって管理される。第2のファイルシステムは、情報処理装置1のOSがFATを介さずにアクセスするためのものであり、OSには予約領域FL-R内の管理、つまり第2のファイルシステムの構築やそれに基づく記録再生アクセス、さらには記録データ、再生データの処理についてアルゴリズムが用意される。特に、予約領域FL-R内では、例えばストリームファイルを構成するデータは、物理空間的に連続するブロックに順番に記録されていくこと、或いはそのために再生時には物理空間的に連続するブロックが順番にアクセスされていくことが規定されており、そのような連続したブロックに対するアクセス動作のためのアルゴリズムが形成されるものとなる。また後述するが、記録データについてのインターリーブや再生データについてのデインターリーブなどの処理も規定される。

【0097】予約領域FL-R内は、例えば図21、図22のように構成される。図21は、例えばメモリアード70内に予約領域FL-Rが設定された状態を示している。上記の第2のファイルシステムによる管理形態を実現するため、予約領域FL-R内には図示するようにポインタ部とデータ部が形成される。ポインタ部はファイルネームと、そのファイルのスタートアドレスポインタSTPを管理する領域とされる。図21の場合は、まだ何もファイルが記録されていない状態としているため、データ部の全体はフリーエリア（記録可能領域）とされ、ポインタ部には、このフリーエリアに対して仮想的にファイル名FL-R-freeというファイル名が設定されて、データ部の先頭アドレスがスタートアドレスポインタSTP-1として示されている。これは、スタートアドレスポインタSTP-1で示されるブロックから、物理的に連続したブロックに順番にデータ記録を実行していき状態を示しているものとなる。

【0098】図22は、予約領域FL-Rに2つのファイルFL-R-1、FL-R-2が記録された状態を示している。ポインタ部においては、ファイルネームFL-R-1に対応して、ファイルの先頭（この場合データ部の先頭）を示すアドレスがスタートアドレスポインタSTP-1として示されている。またファイルネームFL-R-2に対応して、ファイルFL-R-2の先頭を示すアドレスがスタートアドレスポインタSTP-2として示されている。またフリーエリアの先頭として、ファイル名FL-R-freeに対応してスタートアドレスポインタSTP-3として示されている。

【0099】ここでファイルFLR-1、FLR-2は、それぞれ物理空間的に連続したブロックに記録されたもので、これらのファイルが離散的に形成されることはない。

【0100】つまり本例では、メモ리카ード70内に、予約領域FLRを設定し、その予約領域内ではFATではない第2のファイルシステムでファイルが管理されるようにする。また予約領域内に記録されるファイルは、物理空間的に連続するブロックに記録され、離散的に形成されることはない。なお、ファイルを構成するデータは、データストリームがそのままでのデータ順で記録されるようにしてもよいが、所定単位でインターリーブがかけられて、データ順が変更されるようにしてもよい。あるいはデータが暗号化されたものであってもよい。データのインターリーブや暗号化については情報処理装置1のOS上の予約領域管理アルゴリズムによって実行される。

【0101】第2のファイルシステムによる管理形態としてはファイル名に対応してファイルのスタートアドレスポインタSTPを管理するものとする。上述のように各ファイルは物理空間的に連続したブロックに記録されているので、例えば再生の際には、スタートアドレスポインタSTPを参照すれば、後は順にブロックをアクセスしていけばよいものとなる。なお、図示していないが、ファイルのエンドアドレスを管理するようにしてもよいが、例えば物理空間的に連続してデータが暗号化され、またその記録に応じてフリーエリアが後退していくようにすれば、エンドアドレスは自明であるため管理しなくてもよい。即ち図22の場合、ファイルFLR-1の終端は、ファイルFLR-2の先頭アドレス(STP-2)の1つ前のブロックであり、またファイルFLR-2の終端は、フリーエリアの先頭アドレス(STP-3)の1つ前のブロックであるためである。また、スタートアドレスポインタSTPは、暗号化された値とされており情報処理装置1のOS上の予約領域管理アルゴリズムによってのみ解读できるようにしてもよい。

【0102】このような予約領域FLRについての設定、管理方式、記録/再生動作によれば、次のような利点が得られるものとなる。即ちメモ리카ード70内に、物理的に連続した連続記録空間としての予約領域FLRを確保して、その予約領域FLR内でのデータファイルの記録再生が第2のファイルシステムに管理されるようにすることで、予約領域FLRを、FATによる記録再生領域として用いることができる。そして予約領域FLRは物理空間的に連続した記録空間とされていることで、オーディオデータ、動画ビデオデータなどのストリームデータの記録を、物理的に連続して実行できる。例えば図22のファイルFLR-1、FLR-2などの記録や、その再生の際には、FATのリンクをたどるなどの処理は不要で処理負担が少なくなり、しかも

高速な記録再生アクセスが可能となるため、ストリームデータの記録再生に非常に好適なものとなる。また予約領域FLR自体は、FATにより管理される状態にあることで、メモ리카ード70内での予約領域FLRの位置管理や、他の通常のファイルFL1、FL2、FL3等との整合もとれ、FATの管理上の混乱は生じない。

【0103】また物理空間的に連続したブロックにデータファイルが記録されることで、例えばオーディオデータ等の早送り、早戻し再生などの場合に単純にブロックの前後をたどっていけばよく、非常に容易な処理で早送り、早戻し再生を実現できる。

【0104】またストリームデータのそのままでのデータ順序で、連続したブロックに記録していくようにし、またそれを再生するようにすれば、記録再生処理は非常に容易となる。その一方で、ストリームデータについてインターリーブや暗号化を行なって記録することで、他の記録媒体にコピーされても、オーディオビデオを再生できない状態とすることができ、著作権保護に好適である。実際には、例えば著作権保護が必要なデータについては例えばインターリーブしたデータを記録するようにし、著作権保護が不要なデータについてはインターリーブを実行しないで記録するなどの手法がとれらる好適である。

【0105】8. 予約領域生成処理

以上のような予約領域FLRについての、情報処理装置1の予約領域管理アルゴリズムによって実行される設定処理、記録処理、再生処理としての例を、次に述べていく。

【0106】図23はメモ리카ード70に予約領域FLRを設定する処理手順を示すフローチャートである。メモ리카ード70に予約領域FLRを設定する際には、まずステップF101として、装填されているメモ리카ード70からFATを取り出す。そしてステップF102としてデフラグを行い、ステップF103としてメモ리카ード70上で比較的大きな、物理的に連続する空間を確保する。なお、デフラグは必ず実行しなければならないものではない。

【0107】次にステップF104として、確保した連続領域を1つのゲームファイルとして固定する。これは図20に示したように、連続空間が予約領域FLRとしての1つの仮想的なファイルとしてFATから管理されるようにする処理である。即ち確保した連続空間の全ブロックがFAT上で1つのファイルとして管理されるようにFATを更新する処理となる。最後に、ステップF105で、確保した連続空間、即ち予約領域FLRに、第2のファイルシステムを設置する。即ち予約領域FLR内を図21に示した状態となるようにフォーマットする処理である。なお、この場合、予約領域FLRのボイナ部とデータ部が設定されるが、データ部は

すべてフリーエリアとなっている。そしてポインタ部では、フリーエリアのスタートアドレスポインタSTPが記録されることになる。ここでスタートアドレスポインタSTPの値は、所定の暗号化を施すことで、この情報処理装置1でしか認識(解読)できないようにしている。ただし、実際の例としては、スタートアドレスポインタSTPに暗号化を施さない処理例も考えられる。スタートアドレスポインタSTPを暗号化することは、他の情報処理装置や記録再生装置で予約領域FL-Rを正しく使用できない状態とすることを意味し、例えば著作権保護の要望がある場合に有効である。

【0108】以上の処理により、1つの巨大な連続空間が予約領域FL-Rとして設定されたことになり、以後、上述のように予約領域FL-R内へのFATを介さないアクセスが可能となる。

【0109】9. 予約領域へのファイル処理

図24に予約領域FL-Rへのファイルの記録処理のフローチャートを示す。上記のようにして設定されたメモカード70の予約領域FL-Rに対して、例えばオーディオデータファイルや動画ビデオデータファイルの記録を行う場合は、まずステップF201としてFATを参照し、予約領域FL-Rの位置を確認する。

【0110】メモカード70上で予約領域FL-Rの位置を確認したら、その後は第2のファイルシステムに基づいてアクセスを行うことになる。即ちステップF202で予約領域FL-Rのポインタ部からフリーエリアのスタートアドレスポインタSTPを判別する。本例では、スタートアドレスポインタSTPについて暗号化が施されることとしているため、スタートアドレスポインタSTPの値の復号(暗号解読)を行うことになる。

【0111】スタートアドレスポインタSTPからフリーエリアの先頭アドレスを判別されたら、ステップF203で、アクセスするブロックアドレスの値(アクセスアドレスAA)として、解読したスタートアドレスポインタSTPの値をセットする。続いてステップF204、F205、F206、F207のループ処理で、データの記録を実行する。即ちステップF204では記録するデータストリームから1ブロック分のデータを取りだし、そのブロックデータについてインターリーブをかける。そしてステップF205で、インターリーブされたデータをメモカードインターフェース28からメモカード70に転送し、アクセスアドレスAAで示されているアドレスに対して記録させる。ステップF206ではアクセスアドレスAAの値をインクリメントし、連続する次のブロックアドレスを示す値とする。ステップF207において記録するデータが終了したか否か、つまりデータすべての記録が完了したか否かを判断し、終了していなければステップF204に戻る。

【0112】従ってこのステップF204～F207のループでは、記録するデータを1ブロック単位毎にイン

ターリーブをかけ、そのインターリーブされたデータを、フリーエリアの先頭から1ブロックずつ物理的に連続するブロックに記録していく処理が行われることになる。

【0113】記録するファイルについてのすべてのデータが上記ループ処理でメモカード70に記録されたら、ステップF207からステップF208に進み、予約領域FL-Rのポインタ部の更新を行う。即ち、まず記録されたファイル名とともに、そのファイルの先頭アドレスをスタートアドレスポインタSTPとして記録する。この場合、その時点でフリーエリアのスタートアドレスポインタSTPの値が、今回記録されたファイルのスタートアドレスポインタSTPとなる。また、フリーエリアが使用されたことによって、フリーエリアのスタートアドレスポインタSTPの値も更新される。これは、この時点においてアクセスアドレスAAとして保持されている値が、新たなフリーエリアのスタートアドレスポインタSTPの値となる。なお、記録されたファイル及びフリーエリアのスタートアドレスポインタSTPの値は暗号化された値が記録される。

【0114】以上の処理により予約領域FL-Rへのデータファイルの記録が完了する。このような記録処理により、例えば図22に示したようにファイルデータの記録が行われた状態となる。つまり物理的に連続した巨大なエリアである予約領域FL-R内において、1つのファイルが物理的に連続したブロックに記録される状態となる。

【0115】なお、この例では記録データについて1ブロックデータ分の単位でインターリーブをかけるようにしたが、インターリーブを行わない処理も考えられる。また複数ブロック分のデータにわたる範囲でインターリーブをかけてもよい。さらに、インターリーブではなく、記録データに暗号化を施してもよいし、インターリーブと暗号化の両方の処理を施してもよい。

【0116】10. 予約領域からの再生処理

図25に予約領域FL-Rに記録されたファイルの再生処理のフローチャートを示す。例えば図22のように予約領域FL-Rに記録されたファイルに対して、ステップF301として再生要求が発生すると、まずステップF302でFATを参照し、予約領域FL-Rの位置を確認する。

【0117】メモカード70上の予約領域FL-Rの位置を確認したら、その後は第2のファイルシステムに基づいてアクセスを行うことになる。即ちステップF303で予約領域FL-Rのポインタ部から、再生要求のあったファイルのスタートアドレスポインタSTPを判別する。上述のようにスタートアドレスポインタSTPについては暗号化が施されることとしているため、スタートアドレスポインタSTPの値の復号(暗号解読)も行う。

【0118】スタートアドレスポインタSTPから要求されたファイルの先頭アドレスが判別されたら、ステップF304で、アクセスするブロックアドレスの値(アクセスアドレスAA)として、解読したスタートアドレスポインタSTPの値をセットする。続いてステップF305、F306、F307のループ処理で、データの再生を実行する。即ちステップF305では、アクセスアドレスAAのブロックにアクセスし、そのブロックのデータを読み出す。そして上記記録処理ではブロックデータに対してインターリーブがかけられているため、それに対応してデインターリーブを実行し、データ配列を元に戻す。なお、インターリーブがかけられていない場合はこの処理は当然不要である。また暗号化がかけられていれば、暗号解読が必要である。

【0119】次にステップF306ではアクセスアドレスAAの値をインクリメントし、連続する次のブロックアドレスを示す値とする。そしてステップF307において再生するデータが終了したか否か、つまりファイルを構成するデータすべての読み出しが完了したか否かを判断し、終了していなければステップF305に戻る。

【0120】従ってこのステップF305～F307のループでは、物理的に連続するブロック順にデータを順次読み出し、デインターリーブ等の必要な処理を行ってデータを再生していく処理となる。データに暗号化が施されているのであれば、当然暗号解読も行われる。

【0121】再生するファイルについてのすべてのデータが上記ループ処理でメモリカード7から読み出されたら、ステップF307から処理を終了する。なおデータの読み出し終了は、例えばアクセスアドレスAAの値が、他のファイル又はフリーエリアのスタートアドレスポインタSTPの値と同一となったことで判別できる。また、ポインタ部にファイルの終端アドレスが管理されていれば、その終端アドレスの値とアクセスアドレスAAの比較により判別できる。

【0122】以上の処理により予約領域FL-Rからのデータファイルの再生が実行される。このような再生処理では、物理空間的に連続した巨大なエリアである予約領域FL-R内において、1つのファイルを構成するデータが物理空間的に連続したブロックから順に読み出されていくことになり、FATのリンク管理を参照するような処理は不要である。従って上述したように処理負担の軽減、高速再生が可能となり、また図25には示していないが、早送り、早戻し再生などの要求があった場合も、アクセスアドレスAAの前又は後方への更新のみで対応できるため、非常に有効なものとなる。また早送り、早戻し再生のために大容量のキャッシュメモリを備える必要もない。

【0123】以上、実施の形態としての情報処理装置の構成、記録媒体の例、予約領域FL-Rに関する設定、記録、再生処理を説明してきたが、本発明はこれらの例

に限定されることなく、各種の変形例が考えられる。例えば予約領域FL-Rにおける第2のファイルシステムによる管理形態、予約領域FL-Rのエリア設定、設定、記録、再生のための処理手順などは多様な変形例が考えられる。また予約領域FL-Rに記録するファイルはオーディオデータファイル等のストリームファイルが好適であるが、テキストファイルやアプリケーションプログラムファイル等の記録/再生にも当然利用できるものである。また本発明を適用できる装置は、携帯型の情報処理装置のみでなく、多岐にわたるものである。

【0124】

【発明の効果】以上の説明から理解されるように本発明によれば、記録媒体内に、物理的に連続した連続記録空間を確保して、その連続記録空間内でのデータファイルの記録再生が第2のファイルシステムに管理されるようにすることで、連続記録空間を、FAT等の第1のファイルシステムによらない記録再生領域として用いることができる。そして連続記録空間は物理的に連続した記録空間とされていることで、オーディオデータ、動画ビデオデータなどのストリームデータの記録を、物理的に連続して実行できる。これにより、FATのリンクをたどるなどの処理は不要で処理負担が少なくなり、しかも高速な記録再生アクセスが可能となるため、ストリームデータの記録再生に非常に好適なものとなる。また、連続記録空間自体は、第1のファイルシステムにより管理される状態にあることで、記録媒体内での連続記録空間の設定位置は良好に管理でき、また第1のファイルシステムで管理される通常のファイルの記録位置との整合も簡単にとれ、例えば管理上の混乱は生じない。

【0125】また第2のファイルシステムは、連続記録空間に記録されるデータファイルについての先頭位置情報を管理するとともに、連続記録空間に記録されるデータファイルは、物理空間的に連続した領域に所定の規則的処理によるデータ順序で記録されるようにすることで、連続記録空間に記録されるファイルは、先頭位置情報を参照するのみで、記録媒体上のブロックを連続してアクセスしていくことができ、上述した、第1のファイルシステムを参照しないアクセスが容易に実現できるものとなる。また物理空間的に連続した領域にデータファイルが記録されることで、例えばオーディオデータ等の早送り、早戻し再生などの場合に単純にブロックの前後をたどっていけばよく、非常に容易な処理で早送り、早戻し再生を実現できる。

【0126】また上記の所定の規則的処理としては、ストリームデータのそのままのデータ順序とすれば、記録再生処理は非常に容易となる。その一方で、規則的処理としてインターリーブや暗号化を行うことで、他の記録媒体にコピーされても、オーディオやビデオを再生できない状態とすることができ、著作権保護に利用できる。

【図面の簡単な説明】

【図1】本発明の実施の形態の情報処理装置の平面図、右側面図、左側面図、上面図である。

【図2】実施の形態の情報処理装置のブロック図である。

【図3】実施の形態の情報処理装置のOS構造の説明図である。

【図4】実施の形態の情報処理装置で扱うデータベース構造の説明図である。

【図5】実施の形態のメモリカードの外形状を示す平面図、正面図、側面図、底面図である。

【図6】実施の形態のメモリカードの内部構造の説明図である。

【図7】実施の形態のファイルシステム処理階層の説明図である。

【図8】実施の形態のメモリカードの物理的データ構造の説明図である。

【図9】実施の形態のメモリカードの管理フラグの説明図である。

【図10】実施の形態のメモリカードにおけるデータ更新処理と物理アドレス及び論理アドレスの概念の説明図である。

【図11】実施の形態の論理-物理アドレス変換テーブルの管理形態の説明図である。

【図12】実施の形態の論理-物理アドレス変換テーブルの構造の説明図である。

【図13】実施の形態のメモリカードのフラッシュメモリ容量/ブロック数/1ブロックの容量/1ページの容量/論理-物理アドレス変換テーブルのサイズの関係の説明図である。

【図14】実施の形態のメモリカードのディレクトリ構造の説明図である。

【図15】FAT構造の説明図である。

【図16】FATによるクラスタ管理形態の説明図である。

【図17】ディレクトリの内容の説明図である。

【図18】サブディレクトリ及びファイルの格納形態の説明図である。

【図19】実施の形態の情報処理装置とメモリカードのインターフェース構成の説明図である。

【図20】実施の形態の予約領域の説明図である。

【図21】実施の形態の生成された予約領域内の説明図である。

【図22】実施の形態のファイルが記録された予約領域内の説明図である。

【図23】実施の形態の予約領域の生成処理のフローチャートである。

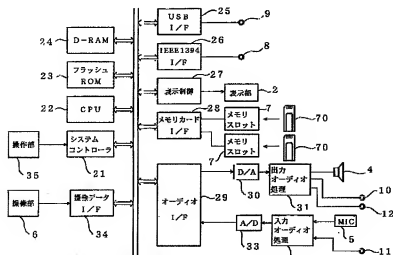
【図24】実施の形態の予約領域への記録処理のフローチャートである。

【図25】実施の形態の予約領域からの再生処理のフローチャートである。

【符号の説明】

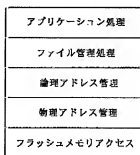
1 情報処理装置、2 表示部、3a、3b、3c 操作部、4 スピーカ、5 マイクホン、6 撮像部、7 メモリスロット、8 IEEE1394端子、9 USB端子、10 ヘッドホン端子、11 ライン入力端子、12 ライン出力端子、21 システムコントローラ、22 CPU、23 フラッシュROM、24 D-RAM、25 USBインターフェース、26 IEEE1394インターフェース、27 表示制御部、28 メモリカードI/F、29 オーディオI/F、30 D/A変換部、31 出力オーディオ処理部、32 A/D変換部、33 入力オーディオ処理部、34 通信インターフェース、35 操作部、36 撮像部

【図2】



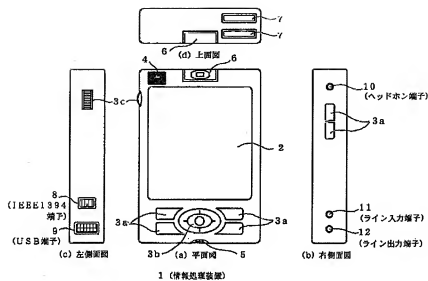
1 (情報処理装置)

【図7】

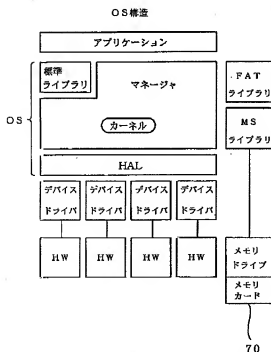


ファイルシステム処理階層

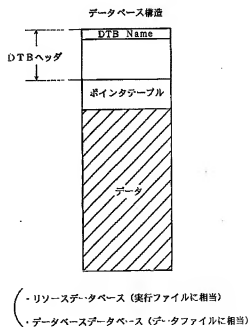
【図1】



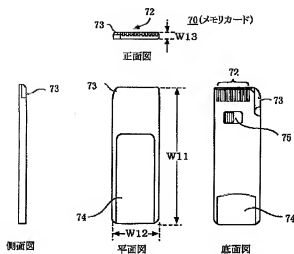
【図3】



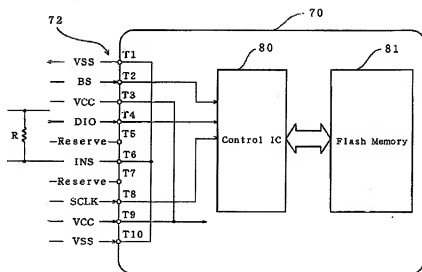
【図4】



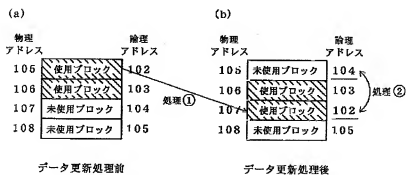
【図5】



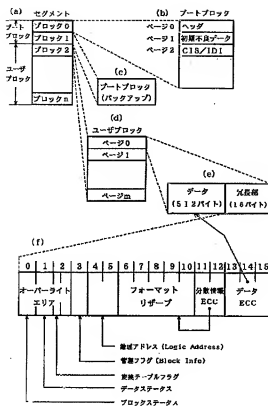
【図6】



【図10】



【図8】



【図9】

管理フラグ

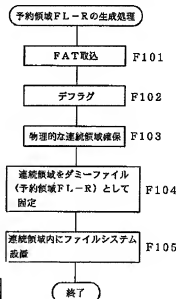
ビット	定義
7	リザーブ
6	リザーブ
5	アクセス許可 (1: free 0: Read Protected)
4	コピー禁止指定 (1: OK 0: NG)
3	変換テーブルフラグ (1: 無効 0: テーブルブロック) *最終セグメントのみ有効
2	システムフラグ (1: ユーザブロック 0: ブートブロック)
1	リザーブ
0	リザーブ

【図11】

物理アドレス (2Byte)	論理アドレス (2Byte)
0x00	0x03 ← 論理アドレス 0x0000
0x00	0x04 ← 論理アドレス 0x0001
0x00	0x04 ← 論理アドレス 0x0002
0x00	0x05 ← 論理アドレス 0x0003
0x01	0xA8 ← 論理アドレス 0x0004
0x00	0x06 ← 論理アドレス 0x000b

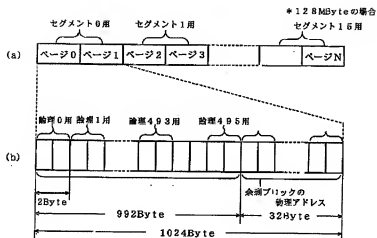
論理アドレス順に、対応する物理アドレスを格納する。

【図23】



【図12】

論理／物理アドレス変換テーブル



【図13】

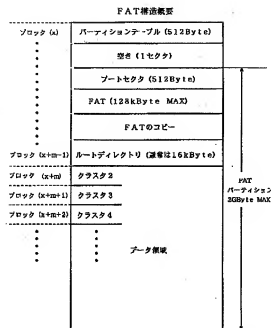
ブッシュメモリ 容量	ブロック数	1ブロックの 容量	1ページの 容量	論理／物理アドレス 変換テーブルサイズ
4KB	512 (1セグメント)	8KB (16page)	(512+16) B	1KB (2page)
8KB	1024 (2セグメント)	8KB (16page)	(512+16) B	2KB (4page)
16KB	2048 (4セグメント)	8KB (16page)	(512+16) B	4KB (8page)
	1024 (2セグメント)	16KB (32page)	(512+16) B	2KB (4page)
32KB	2048 (4セグメント)	16KB (32page)	(512+16) B	4KB (8page)
	4096 (8セグメント)	16KB (32page)	(512+16) B	8KB (16page)
128KB	8192 (16セグメント)	16KB (32page)	(512+16) B	16KB (32page)

【図14】

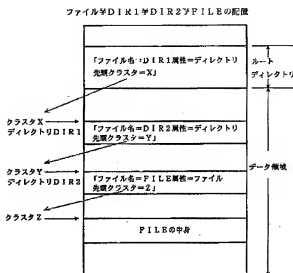
ディレクトリの構成

ROOT	VOICE	音声用ディレクトリ
	DLIM	静止画用ディレクトリ
	MOxxxxnn	動画用ディレクトリ
	CONTROL	制御用ディレクトリ
	HIFI	音楽用ディレクトリ
	PM	情報処理装置用ディレクトリ

【図15】



【図18】



【図16】

FAT概念図

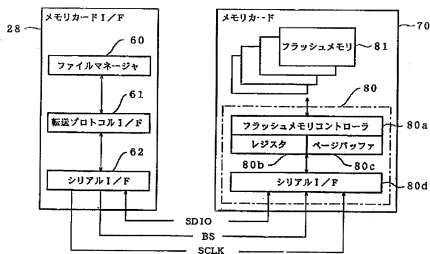
	クラスター番号	FAT	クラスター
ディレクトリ	002	003	MAIN.Cのデータ1
MAIN.C 002	003	006	MAIN.Cのデータ2
FUNC.C 004	004	005	FUNC.Cのデータ1
	005	FFF	FUNC.Cのデータ2
	006	FFF	MAIN.Cのデータ3
	007	000	未使用*
	008	000	未使用*

【図17】

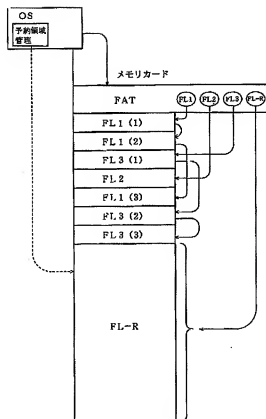
ファイル名1本分のディレクトリの構造。()はバイト数。

ファイル名(8)	拡張子(3)	属性(1)	リザーブ(1)	時刻(2)	日付(2)	先頭クラスター(2)	サイズ(4)
----------	--------	-------	---------	-------	-------	------------	--------

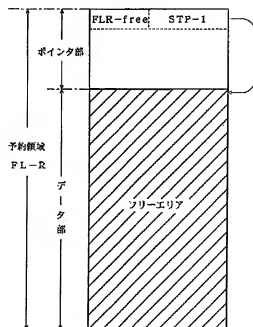
【図19】



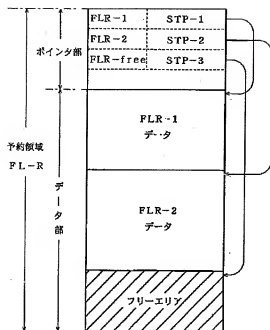
【図20】



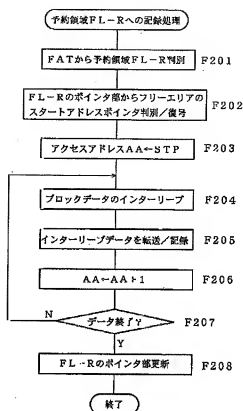
【図21】



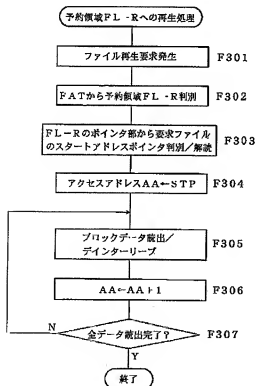
【図22】



【図24】



【図25】



(特4) 101-325128 (P2001-325128A)

フロントページの続き

(72)発明者 田平 武彦

Fターム(参考) 5B082 BA01 CA03 EA04

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内